LightWave New Feature Roundup

NuWave

Product Releases

- Maintenance Releases
  - Last week of February and August
  - Primarily bug fixes
  - May contain minor, low risk new features
- Feature Releases
  - Between maintenance releases, flexible schedule
  - Includes alpha / beta releases
  - Major new functionality

Product Releases

- Hotfixes
  - Critical issues impacting production.
  - Minimal change to currently running release

Product Releases

- Fixes and new features documented in release notes
- Hotfixes documented on support site under “Product Advisories”
- Announcements made via email blast. If you aren’t receiving the announcements and want to, let us know.

Product Releases

Version string format: 1.2.3[.4]

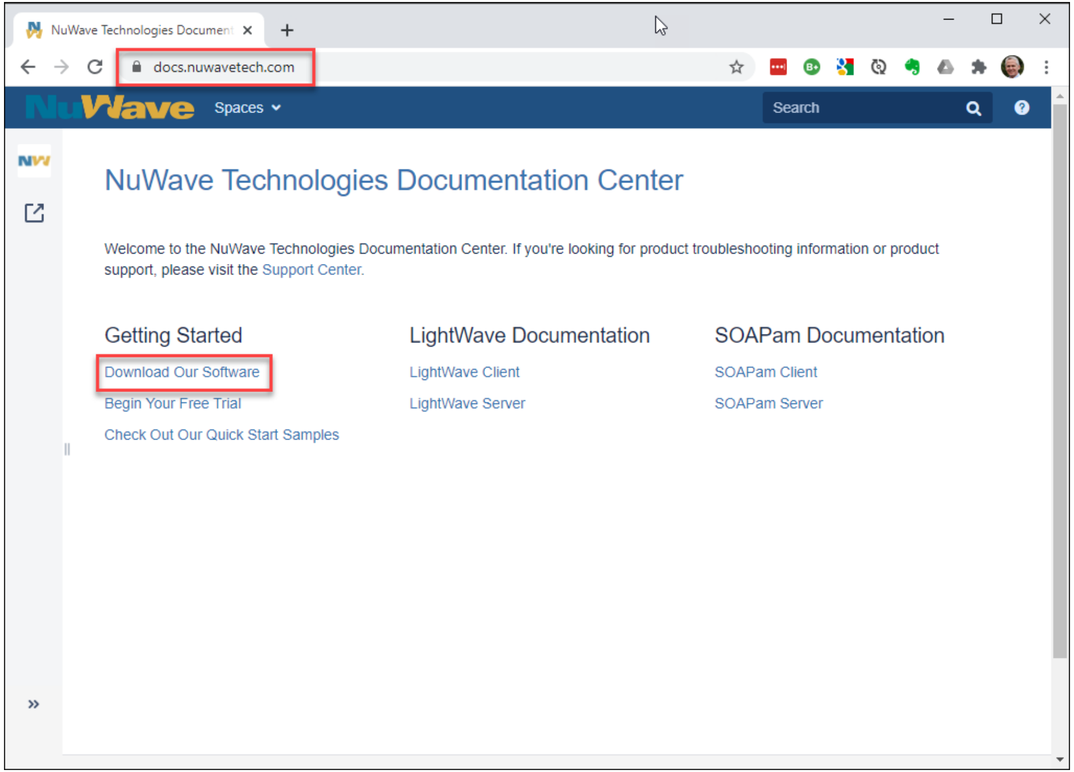
1: major version

2: minor version - new feature release

3: patch version – maintenance release

4: optional hotfix version

Product Releases



NuWave Technologies Documentation Center

Welcome to the NuWave Technologies Documentation Center. If you're looking for product troubleshooting information or product support, please visit the Support Center.

Getting Started

- Download Our Software
- Begin Your Free Trial
- Check Out Our Quick Start Samples

LightWave Documentation

- LightWave Client
- LightWave Server

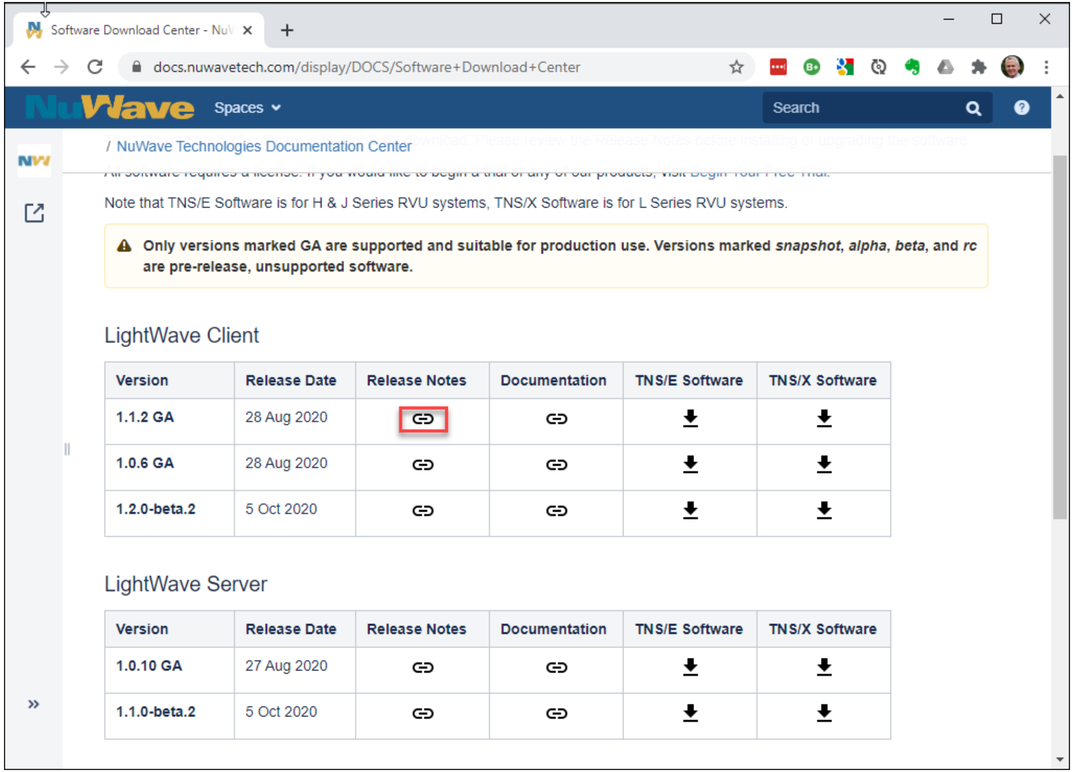
SOAPam Documentation

- SOAPam Client
- SOAPam Server

docs.nuwavetech.com

NuWave

Product Releases


Software Download Center - NuW
docs.nuwavetech.com/display/DOCS/Software+Download+Center
NuWave
Spaces
Search
NW

/ NuWave Technologies Documentation Center

All software requires a license. If you would like to begin a trial of any of our products, visit Begin Your Free Trial.

Note that TNS/E Software is for H & J Series RVU systems, TNS/X Software is for L Series RVU systems.

Only versions marked GA are supported and suitable for production use. Versions marked *snapshot*, *alpha*, *beta*, and *rc* are pre-release, unsupported software.

LightWave Client

Version	Release Date	Release Notes	Documentation	TNS/E Software	TNS/X Software
1.1.2 GA	28 Aug 2020	🔗	🔗	⬇	⬇
1.0.6 GA	28 Aug 2020	🔗	🔗	⬇	⬇
1.2.0-beta.2	5 Oct 2020	🔗	🔗	⬇	⬇

LightWave Server

Version	Release Date	Release Notes	Documentation	TNS/E Software	TNS/X Software
1.0.10 GA	27 Aug 2020	🔗	🔗	⬇	⬇
1.1.0-beta.2	5 Oct 2020	🔗	🔗	⬇	⬇

Working with Optional Elements

- Prior to Feb 2020 releases it was difficult to indicate or determine that a message element was optional.
- Both products now support schema attributes / properties:
  - hideIfEmpty
  - isSet
  - minOccurs (deprecated)

Working with Optional Elements

hideIfEmpty -

- May be set on any element to indicate that if the source element is *empty*, it should not be included when serializing a request.
- An element is considered empty when it is completely filled with the element's string padding character.
- Good choice when fields or structures are space or null filled.

Working with Optional Elements

isSet –

- May be set on any element, and references a target integer element which is used to indicate the presence or absence of the source element.
- Can be used to determine if the inbound message contains an element.
- Good choice when element contains mixed data with spaces, null, or numeric values.

Working with Optional Elements

minOccurs –

- The *minOccurs="0"* attribute may be set on elements with *type="string"* to indicate that if the source element is filled with the string padding character, it should not be included in the request payload.
- Deprecated … still works but new definitions should use *hideIfEmpty*.

Working with Optional Elements

- Available since:
  - LWS 1.0.9
  - LWC 1.1.1
  - LWC 1.0.5
- Tutorial available on YouTube:
  - Youtube.com/user/nuwavetech
  - Or, search for “lightwave optional elements”

NuWave

Substitution Variables

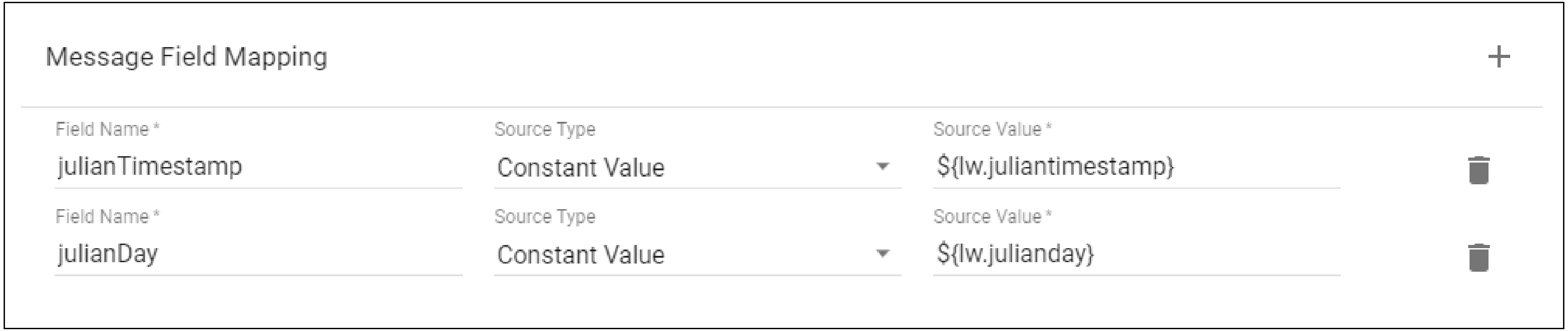
- LWS 1.0.9+ can inject runtime data into the request IPM.
- When a message field mapping source type is Constant, a substitution variable may be supplied as the constant value. The field will be set with the corresponding value of the substitution variable.

Substitution Variables

- Available values are:
  - ${lw.juliantimestamp} - The current julian timestamp value.
  - ${lw.julianday} - The current julian day value.
  - ${lw.timestamp48} - The current 48 bit timestamp.

- Adding more substitutions in next releases (unique id, client/server IP, etc.). Suggestions welcome…

Substitution Variables



Message Field Mapping

Field Name *	Source Type	Source Value *
julianTimestamp	Constant Value	${lw.juliantimestamp}
julianDay	Constant Value	${lw.julianday}

LWC Outbound IP Address Binding

- Option *tcpip-bind-addr*, specifies an IPv4 address to bind to for TCP/IP connections. This option may be used to specify the IP address from which connections will originate when a TCPIP provider is configured with multiple IP addresses.
- Available in LWC 1.1.2+.

Measure Counters

- LWC has supported measure counters since 1.0.2, Apr 2018
- LWS has supported measure counters since 1.0.7, Oct 2018

Developer Guide | Advanced Topics | Using MEASURE Counters

New Stuff

- LightWave Server 1.1.0
- LightWave Client 1.2.0

TLS Features

- TLS 1.3, based on OpenSSL 1.1.1
    - TLS 1.0 & TLS 1.1 disabled by default, but can be enabled
- By the way, existing options to disable older TLS versions:
    - --tls-disable-v1.0
    - --tls-disable-v1.1
- SSL not supported

BLOB Support in LWS

- When using BLOBs (Binary Large Objects), your application provides or receives the Web service payload as raw data in the IPM or in an Enscribe file.
- When a file is used, the name of the file is supplied in the request and/or response IPMs.

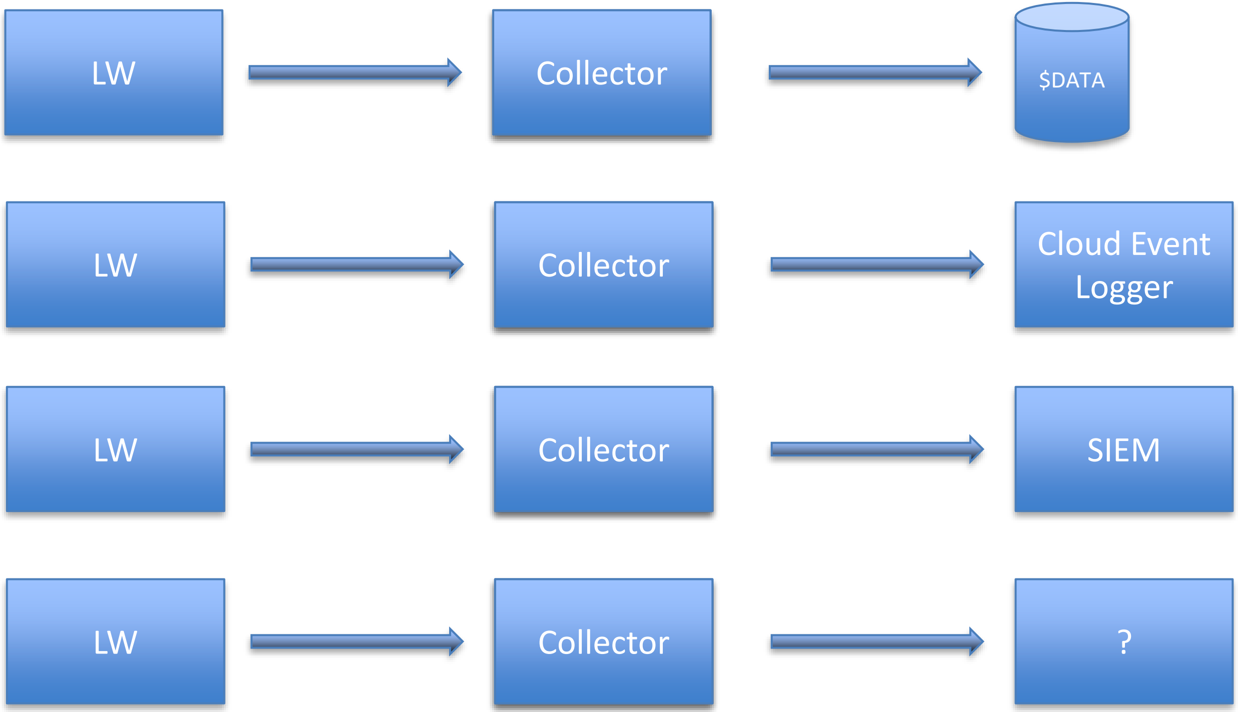
BLOB Support in LWS

- Currently being retooled based on Beta feedback.
- If you've seen the YouTube video, the basic functionality remains the same, but configuration has been simplified.
- Stay tuned …

Message Logging

- Provides the same information available in diagnostic logs, but in a fast, flexible, customizable format.
- You configure rules to select requests to log, and what data you want from each request, LightWave delivers it to your custom Pathway server as a structured IPM.
- You do whatever you want with it.

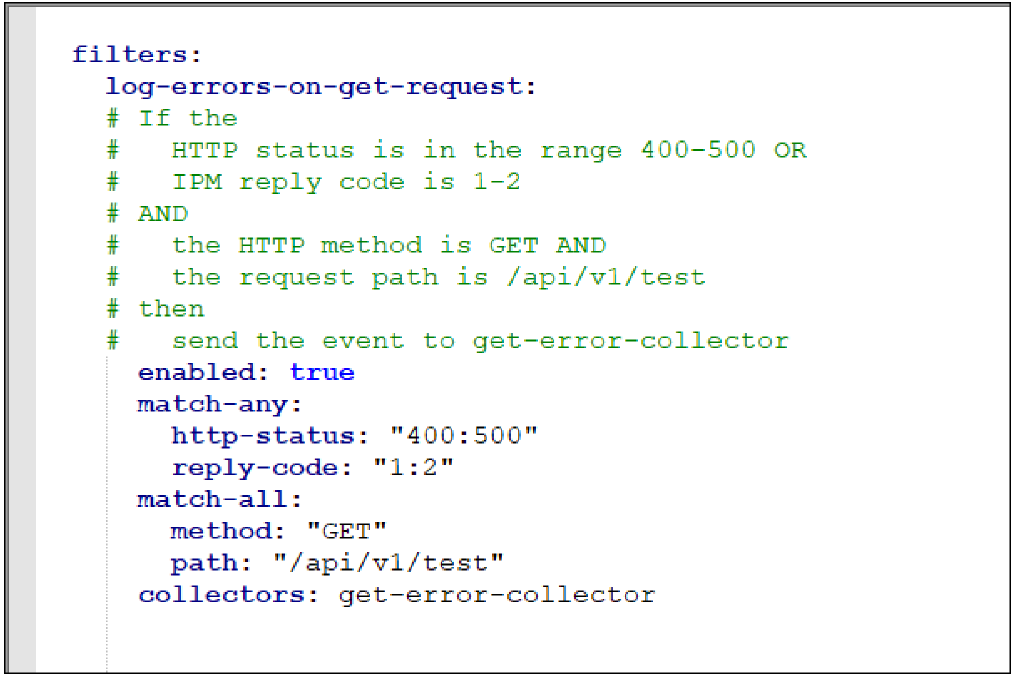
Message Logging



Message Logging

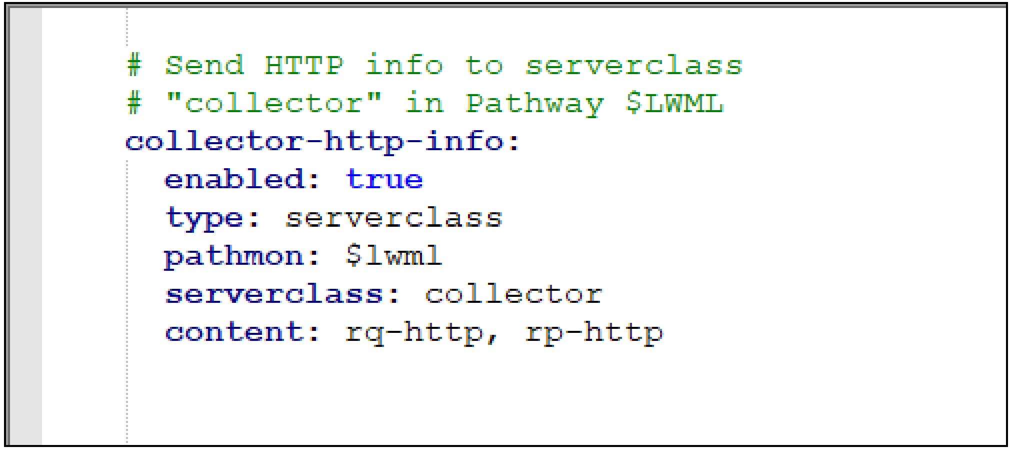
- Uses YAML for configuration
- YAML validator built into LWSCOM & LWCCOM
- Flexible rules-based message selection
- Flexible content delivery options

Message Logging



```
filters:
  log-errors-on-get-request:
  # If the
  #   HTTP status is in the range 400-500 OR
  #   IPM reply code is 1-2
  # AND
  #   the HTTP method is GET AND
  #   the request path is /api/v1/test
  # then
  #   send the event to get-error-collector
    enabled: true
    match-any:
      http-status: "400:500"
      reply-code: "1:2"
    match-all:
      method: "GET"
      path: "/api/v1/test"
    collectors: get-error-collector
```

Message Logging



```
# Send HTTP info to serverclass
# "collector" in Pathway $LWML
collector-http-info:
  enabled: true
  type: serverclass
  pathmon: $lwml
  serverclass: collector
  content: rq-http, rp-http
```

Message Logging

- Tutorial available on YouTube:
  - Youtube.com/user/nuwavetech
  - Or, search for “lightwave message logging”

- Sample collectors in C and COBOL on Github
  - github.com/nuwavetech

Thanks!

Questions?